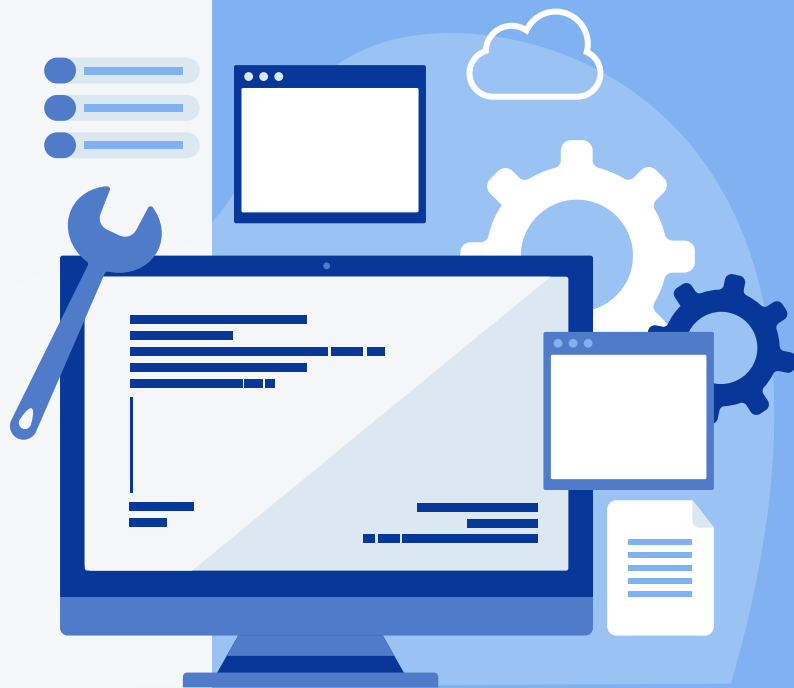


# Programação Web

I

## CONCEITOS BÁSICOS DE CSS

Prof. Aguinaldo Alves





# Apresentação

## SOBRE O PROFESSOR

Me chamo Aguinaldo Alves Pinto, sou formado em Sistemas de Informação, com Especialização em Engenharia de Sistemas e Mestre em Modelagem Matemática e Computacional.

# CSS

## Diferentes Tipos de Seletores

Existem diversos tipos de seletores diferentes, vamos conhecer os seletores de elementos. Que selecionam todos os elementos de um determinado tipo nos documentos **HTML**.

Porém é preciso fazer seleções mais específicas, veja alguns dos tipos mais comuns de seletores.

# CSS

## Diferentes Tipos de Seletores

Nome do seletor	O que seleciona	Exemplo
Seletor de Elemento (pode ser chamado tag ou seletor de tipo)	Todos os elementos HTML de determinado tipo.	P Seleciona <p>
Seletor de ID	O elemento na página com o ID especificado. Em uma determinada página HTML, é uma boa prática usar um elemento por ID (e claro, um ID por elemento) mesmo que seja permitido usar o mesmo ID para vários elementos.	#id Seleciona <p id="meu id"> ou <a id="meu-id">.

# CSS

## Diferentes Tipos de Seletores

Nome do seletor	O que seleciona	Exemplo
Seletor de Classe	O(s) elemento(s) na página com a classe especificada (varias instancias de classe podem aparecer em uma página)	<b>.class</b> Seleciona <code>&lt;p class="minha-classe"&gt;</code> e <code>&lt;a class="minha-classe"&gt;</code>
Seletor de Atributo	O(s) elemento(s) na página com o atributo especificado.	<b>Img[src]</b> Seleciona <code>&lt;img src="minha_imagem.png"&gt;</code> . Lembrando seleciona o atributo <code>src</code> e não o <code>img</code>

# CSS

## Diferentes Tipos de Seletores

Nome do seletor	O que seleciona	Exemplo
Seletor de Pseudo-classe	O(s) elemento(s) especificado(s), mas somente quando estiver no estado especificado, como por exemplo o mouse sobre ele.	<b>a:hover</b> Seleciona <a>, mas somente quando o mouse está em cima do link.

# CSS

## Fontes e Textos

No **CSS** podemos vincular fontes em diferentes formas, uma delas é um **vínculo online** de fonte, nesse caso usando o **google fonts** podemos inserir uma ou mais famílias de fontes com esse método , vamos ver no código:

```
<link href="http://fonts.googleapis.com/css?family=Open+Sans" rel="stylesheet">
```

# CSS

## Fontes e Textos

Nota-se no código acima que vinculamos uma **folha de estilo** que baixa a família de fontes **Open Sans** junto com a página web, permitindo que seja definido seus elementos **HTML** usando a própria folha de estilos. No **CSS** é possível estilizar várias propriedades de um mesmo elemento de texto, vamos conhecer agora esses elementos:

```
<link href="http://fonts.googleapis.com/css?family=Open+Sans" rel="stylesheet">
```



# CSS

## Diferentes propriedades de texto

PROPRIEDADE FONTE	VALOR DA PROPRIEDADE
Font	font-style   font-variant   font-weight   font-size   font-family   caption icon   menu   message-box   smallcaption   status-bar
Font-size	xx-small   x-small   small   medium   large   x-large   xx-large   smaller larger   inherit   length (%)
Font-size-adjust	None   inherit   number (%)
Font-stretch	normal   wider   narrower   ultra-condensed   extracondensed   condensed   semi-condensed   semiexpanded   expanded   extra- expanded   ultraexpanded   inherit

# CSS

## Diferentes propriedades de texto

PROPRIEDADE FONTE	VALOR DA PROPRIEDADE
Font-style	Normal   italic   oblique   inherit
Font-variant	Normal   small-caps   inherit
Font-weight	Normal   bold   bolder   lighter   100   200   300   400   500   600   700   800   900   inherit
Font-family	Family-name   generic-family   inherit

Na tabela acima estão as **possibilidades** que podemos definir os **estilos das fontes**, importante que na hora da prática você faça testes para descobrir qual melhor estilo para seu documento.

# CSS

## Diferentes propriedades de texto

PROPRIEDADE DE TEXTO	VALOR DA PROPRIEDADE
<b>direction</b>	ltr   rtl   inherit
<b>hanging-punctuation</b>	None   [ start   end   adjacent ]
<b>letter-spacing</b>	normal   length (%)
<b>punctuation-trim</b>	None   [ start   end   adjacent ]
<b>text-align</b>	Start   end   left   right   center   justify
<b>text-align-last</b>	Start   end   left   right   center   justify
<b>text-decoration</b>	none   underline   overline   line-through   blink
<b>text-emphasis</b>	none   [ accent   dot   circle   disc ] [ before   after ]
<b>text-indent</b>	Length (%)

# CSS

## Diferentes propriedades de texto

PROPRIEDADE DE TEXTO	VALOR DA PROPRIEDADE
<b>text-justify</b>	Auto   inter-word   interideograph   inter-cluster   distribute   kashida   Tibetan
<b>text-outline</b>	None   color   length
<b>text-shadow</b>	None   color   length
<b>text-transform</b>	None   capitalize   uppercase   lowercase
<b>text-wrap</b>	Normal   unrestricted   none   suppress
<b>unicode-bidi</b>	Normal   embed   bidioverride
<b>white-space</b>	Normal   pre   nowrap   prewrap   pre-line
<b>white-space-collapse</b>	Preserve   collapse   preserve-breaks   discard
<b>word-break</b>	Normal   keep-all   loose   break-strict   break-all
<b>word-spacing</b>	Normal   length (%)
<b>word-wrap</b>	Normal   break-word

# CSS

## Diferentes propriedades de texto

Na tabela acima é mostrado a **propriedade de texto** com seus respectivos valores, dessa forma é possível entender que cada propriedade tem um **valor** específico com uma **ação** específica. Não é necessário que saiba todas, o importante é usá-las de acordo achar melhor em sua folha de estilo. Por isso é recomendado que faça testes com cada um desses valores de cada propriedade, assim é possível visualizar o efeito de cada uma delas.

# Desenvolvimento e Aplicação

Vamos começar a utilizar o **CSS** na prática para fixar melhor esse conjunto de regras e seu modo de aplicação.

# Desenvolvimento e Aplicação

```
1 <html>
2   <head>
3     <link rel="stylesheet" href="D:\Exercicios Aulas\CSS\estilo.css">
4
5     <title> - Estilos em CSS - </title>
6   </head>
7   <body>
8
9     <h1>Aplicando Folhas de Estilo No HTML</h1>
10
11    
12
13    <p>Assim como o HTML, o CSS não é realmente uma linguagem de programação.
14      Também não é uma linguagem de marcação – é uma linguagem de folhas de estilos.
15      Isso significa que o CSS permite aplicar estilos seletivamente a elementos em documentos HTML.</p>
16
17  </body>
18 </html>
```

# Desenvolvimento e Aplicação

```
1  /**ALTERANDO A COR DE FUNDO**/  
2  html {  
3      background-color: #424142;  
4  }
```

Alterando o background da página:

```
body {  
    width: 600px;  
    margin: 0 auto;  
    background-color: #0e4d58;  
    padding: 0 20px 20px 20px;  
    border: 3px solid black;
```



# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**width: 600px;** — Isso força o corpo a ter 600 pixels de largura.

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**margin: 0 auto; —**

Quando você define dois valores em uma propriedade como **margin** ou **padding**, o primeiro valor afeta a parte superior do elemento e a parte inferior (tornando-os 0 nesse caso).

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**margin: 0 auto; —**

E no segundo valor os lados esquerdo e direito (aqui, auto é um valor especial que divide o espaço horizontal uniformemente entre esquerda e direita).

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**margin: 0 auto; —**

Você também pode utilizar um, três ou quatro valores, como documentado aqui.

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**background-color;** —

Como antes, isso define a cor de fundo do elemento.

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**padding: 0 20px 20px 20px;** — Temos quatro valores definidos no padding, para criar um pouco de espaço em torno do nosso conteúdo.

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**padding: 0 20px 20px 20px;** — Dessa vez, estamos definindo **sem padding** na parte **superior** do corpo, e **20 pixels** no lado esquerdo, na parte inferior e no lado direito.

# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**padding: 0 20px 20px 20px;** — Os valores definem a parte **superior**, o lado **direito**, a parte **inferior** e o lado **esquerdo**, nessa ordem. Como com a **margin**, você também pode usar um, dois, ou três valores.



# Desenvolvimento e Aplicação

Agora para o elemento **<body>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**border: 3px solid black; —**  
Isso simplesmente define uma borda preta sólida de 3 pixels de largura em todos os lados do corpo.

# Desenvolvimento e Aplicação

Alterando as propriedades h1

```
h1 {  
  margin: 0;  
  padding: 20px 0;  
  color: ■#f08902;  
  text-shadow: 2px 2px 2px □black;  
  text-align: center;  
  font-family: Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;  
}
```

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**margin:** Definimos o **margin** do h1 em 0, para tirar os espaços da margem do texto.

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**padding:** O padding no texto é importante para espaçar dos outros elementos da página.

**color:** Definimos a cor do texto.

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-shadow** – Uma propriedade bastante interessante que usaremos aqui é o **text-shadow**, que aplica uma sombra ao conteúdo de texto do elemento.

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-shadow** –

O primeiro valor em **pixel** define o deslocamento horizontal da sombra do texto — até onde ele se move: um valor negativo deve movê-la para a esquerda.

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-shadow** –

O segundo valor em **pixel** define o deslocamento vertical da sombra do texto — o quanto ela se move para baixo, neste exemplo; um **valor negativo** deve movê-la para cima

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-shadow** –

O terceiro valor em **pixel** define o raio de desfoque da sombra — um valor maior significará uma sombra mais borrada.



# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-shadow** –

O quarto valor define a cor base da sombra.

**text-align:** - Essa propriedade centraliza o texto.

# Desenvolvimento e Aplicação

Agora para o elemento **<h1>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**Font-family:** - Definimos uma família de fontes, ele funciona em ordem, nesse caso se a fonte **Cambria** não estiver instalada no computador ou no servidor (caso o site já esteja hospedado), ele define a fonte **Georgia** para uso, e assim sucessivamente.

# Desenvolvimento e Aplicação

Alterando as propriedades img

```
img {  
  display: block;  
  margin: 0 auto;  
}
```

# Desenvolvimento e Aplicação

Agora para o elemento `<img>`. Há algumas declarações aqui, então vamos passar por elas uma a uma:

Com essa propriedade centralizamos a imagem para melhorar a aparência. Nós poderíamos usar novamente o truque **`margin: 0 auto.`**

# Desenvolvimento e Aplicação

Agora para o elemento **<img>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

O elemento **<body>** é em nível de bloco, o que significa que ocupa espaço na página e pode ter **margens e outros valores** de espaçamento aplicados a ele.

# Desenvolvimento e Aplicação

Agora para o elemento **<img>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**Imagens**, por outro lado, são elementos em linha, o que significa que não podem ter margens.

# Desenvolvimento e Aplicação

Agora para o elemento **<img>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

Então, para aplicar **margens a uma imagem**, temos que dar o comportamento de nível de bloco a imagem usando **display: block**.

# Desenvolvimento e Aplicação

Alterando as propriedades p

```
p{  
  margin: 25px 25px;  
  padding: 25px;  
  text-align: justify;  
  font-size: 25px;  
  color: ■white;  
  text-shadow: 2px 2px 1px □black;  
  font-family: Verdana, Geneva, Tahoma, sans-serif;  
}
```



# Desenvolvimento e Aplicação

Nos parágrafos aqui definimos propriedades para melhor alocar o texto na página, nota-se que as propriedades são bem parecidas com as do elemento **<h1>**;

```
p{  
  margin: 25px 25px;  
  padding: 25px;  
  text-align: justify;  
  font-size: 25px;  
  color: ■white;  
  text-shadow: 2px 2px 1px □black;  
  font-family: Verdana, Geneva, Tahoma, sans-serif;  
}
```

# Desenvolvimento e Aplicação

Agora para o elemento **<p>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**margin:** - Definimos o margin com 25 pixels dos lados, para melhorar e espaçar o texto.

**padding:** - Com o padding deixamos um espaço interno entre o bloco do texto.

# Desenvolvimento e Aplicação

Agora para o elemento **<p>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**text-align** – Aqui nessa propriedade definimos que o parágrafo seja justificado.

**font-size** – Com a font-size deixamos a fonte do parágrafo com tamanho de 25 pixels.

# Desenvolvimento e Aplicação

Agora para o elemento **<p>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

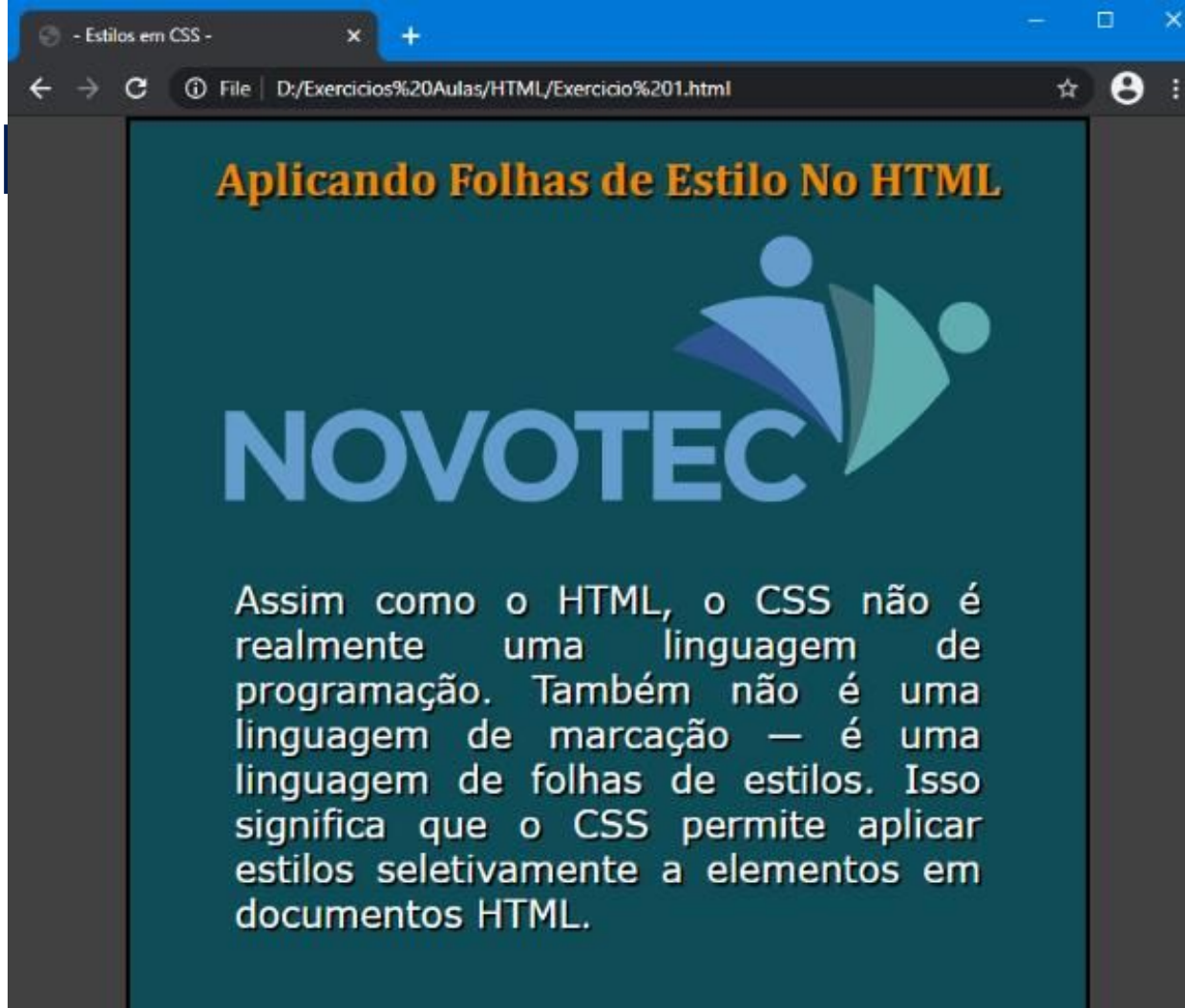
**color:** - Mudamos a cor do texto para branco, dando destaque no texto e contrastando com o fundo.

**text-shadow:** - Colocamos uma sombra em preto no texto para dar um contraste com a cor branca.

# Desenvolvimento e Aplicação

Agora para o elemento **<p>**. Há algumas declarações aqui, então vamos passar por elas uma a uma:

**font-family** – Definimos a família de fontes para nosso parágrafo, assumindo a ordem de importância em caso a fonte não esteja instalada, automaticamente assume a próxima fonte da sequência.



# Exercícios de Fixação

Faça a atividade interativa fixando os conceitos trabalhados até aqui.



genially

Atividade interativa : [Clique aqui](#)

Curso CSS do W3C Brasil: [Clique aqui](#)

# Lista de Exercício VI

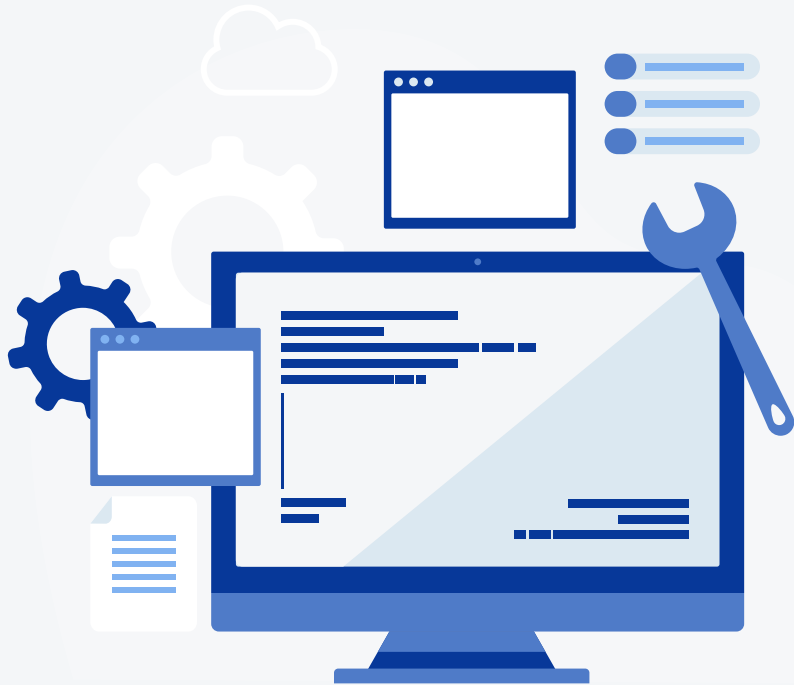
Faça a atividade respondendo às questões propostas (manuscrito e individual).

Visto na próxima aula !!!

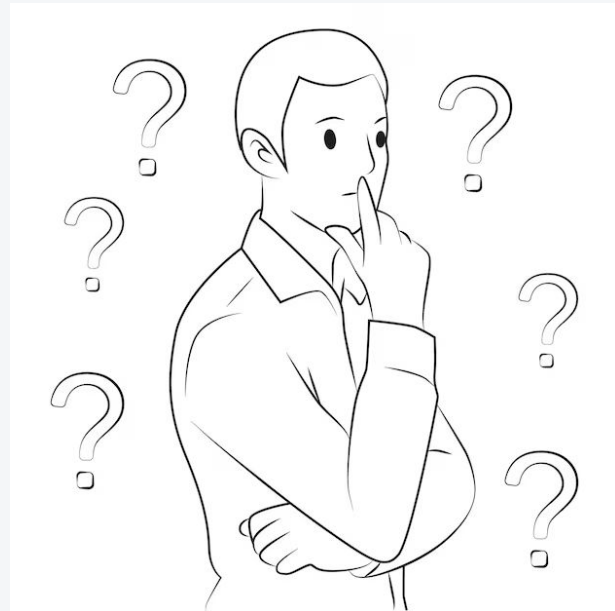
- 1) Quais os principais tipos de seletores em CSS?
- 2) Qual propriedade CSS preciso usar para mudar a cor do texto de um elemento <p> do HTML?
- 3) Qual a diferença entre margin e padding?
- 4) Explique qual o procedimento para incorporar uma fonte disponibilizado pelo google fonts em uma página web.
- 5) Explique as propriedades que podem ser aplicada em uma borda ( border ).



# Obrigado



**Você tem alguma pergunta?**



# Recursos alternativos

Esta é uma seleção de recursos alternativos cujo estilo combina com o deste modelo:

## Vetores

- [Desktop and smartphone app development](#)

